ICIC Express Letters
Part B: Applications
Volume **13**, Number **5**, May **2022**

ICIC International ©2022 ISSN 2185-2766
pp. **519–526**

OBTAINING LIGHTWEIGHT HUMAN ACTIVITY RECOGNITION MODEL THROUGH KNOWLEDGE DISTILLATION OF DEEP NEURAL NETWORK

Muhammad Hanif Ramadhan, Taufik Nur Adi, Hyerim Bae and Hyemee Kim

Industrial Data Science and Engineering, Department of Industrial Engineering
Pusan National University
2, Busandaehak-ro 63beon-gil, Geumjeong-gu, Busan 46241, Korea
{ mhanifr; taufiknuradi; hrbae; khm0219 }@pusan.ac.kr

Received September 2021; accepted November 2021

Abstract. *The application of sensors on mobile and wearable devices has prompted studies seeking to leverage human activity recognition (HAR) sensor data. Deep neural networks have been shown to be a promisingly accurate HAR-inference method; however, deploying such deep models to mobile and wearable devices is difficult given their size and high computational cost. This study, by using knowledge distillation training, attempted to obtain a lightweight HAR model with a compressed knowledge of a deep inception-residual network (deep inception-resnet) model. To train and test the models, Skoda Dataset benchmark was employed. The deep inception-resnet teacher model proposed in this paper was able to obtain a macro-averaged F1-score performance comparable to that of the previous, state-of-the-art model. We then distilled the knowledge of the teacher model into a simple CNN model and obtained an F1-score accuracy of 96.4%, which is a 10% improvement on the baseline CNN model that was trained from scratch. Therefore, with knowledge distillation on the Skoda Dataset benchmark, this study was able to train a simple CNN model with a number of parameters 24.5 times smaller than that of the teacher model and yet achieved performance relatively close to its teacher and the previous state-of-the-art model.*

Keywords: Deep learning, Knowledge distillation, Human activity recognition, Residual networks, Inception networks

1. Introduction.

The application of sensors to mobile and wearable devices has inspired studies to further develop the technologies that can leverage human activity recognition (HAR) sensor data [1,2]. HAR is a ubiquitous and important field of study, as it potentiates a wide range of applications to diverse problems ranging from healthcare to sports and fitness and to manufacturing [3-5]. HAR, for example, can potentially help physicians to monitor and manage the activities of patients with chronic maladies such as cardiovascular disease and diabetes [3]. A self-tracking software to measure physical indicators and provide data on performance in sports or daily activities is an additional known application of HAR [4]. Yet another potential application is the usage of HAR to automatically record the time and motions from operational activities of human operators in manufacturing systems rather than manually gathering them directly on the production floor [5]. Given HAR's extraordinarily wide range of applications, many have attempted to build an inference model to render it automatically [1,2,6-8].

Many of the previous studies on HAR relied on feature engineering to extract raw sensor data into a set of statistical features that are easier to learn by classifiers [9,10]. One of them utilized various statistical features including the mean, standard deviation, feature average of peak frequency (APF), and its variance (VarAPF) to extract features

DOI: 10.24507/icicelb.13.05.519

from each window of the accelerometer data. The extracted features were then utilized to train an ensemble model that yielded an accuracy of 91.15% [9]. Another study extracted three inertial sensors into time-domain and frequency-domain features that were then similarly fed into several classifiers including SVM, Gaussian mixture model (GMM), k-nearest neighbor (k-NN), and hidden Markov model (HMM). Given the feature extraction method, k-NN and HMM were found to be the best for supervised and unsupervised classification, respectively [10].

Whereas previously we established how feature engineering was able to show decent results, the method employed is problematic, as a set of well-designed features in one HAR problem does not necessarily generalize well to other HAR problem cases [11]. This is due to the fact that domain knowledge of the specific HAR cases is often needed when designing such engineered features [2]. One alternative approach that has been taken in recent studies, therefore, is to directly learn the extracted features with deep learning [1,2,6-8,12]. A deep CNN with long short-term memory (DeepConvLSTM) model was utilized by [2] to perform HAR on OPPORTUNITY and the Skoda Datasets. Due to the LSTM ability to capture long-term sequences, their method was able to consistently achieve better performance against CNN, k-NN, and linear discriminant analysis (LDA). One of the best models we have found to date in the HAR literature is a self-attention-based neural network model [8] that performed better than either ConvLSTM [2] or convolution auto-encoder (ConvAE) [13].

One common trend observed in the natural language processing and image recognition fields is the increase of model performance based on deeper and larger architecture [14,15]. This pattern has started to emerge in HAR, as seen in [1] and [7]. However, whereas it is desirable to have a more accurate model for HAR, large and deep models are difficult to implement to commercial devices in real time, due to model size and computational cost. One idea to overcome this obstacle is to perform compression to find a smaller model that performs comparably to the deeper model. One study came up with a simple yet effective way to perform model compression: knowledge distillation in a neural network [16]. With knowledge distillation, it is possible to obtain a small neural network called the "student" by training it along with the larger "teacher" network in addition to the supervised labels. By using MNIST and JFT datasets, [16] showed that a neural network trained with the teacher network performs considerably better than one that was trained from scratch by just hard-coded supervised labels.

Given the problem of improving model accuracy and the related issues of model size and computational cost, the present study aimed to obtain a lightweight sensor-based HAR model with distilled knowledge of a larger and deeper neural network. Specifically, we investigated the effect of knowledge distillation of a deep inception-residual network (deep inception-resnet) to a simple CNN for HAR. We performed the experimentations by using the Skoda Dataset benchmark which is a public dataset obtained by tracking a series of activities performed by a worker in a car maintenance scenario [17]. To further evaluate the result, we also compared the F1-score performance of the model with a baseline CNN trained from scratch without any knowledge distillation.

The present paper is organized into 4 sections with the rest of the sections described as follows. Section 2 briefly introduces the concept of knowledge distillation and discusses in detail how this study designs the architecture for the teacher and the student networks. Section 3 lays out the data preprocessing procedure as well as the training and evaluation of the models. Section 4 discusses the conclusions and the future research directions of this study.

2. Knowledge Distillation.

The model proposed in this paper was designed based on knowledge distillation training [16] consisting of two models: teacher and student networks. The main idea behind knowledge distillation is to let the student network learn from a "softer" probability distribution instead of the "hard" coded supervised labels. In order to formalize the notion of soft distribution and to apply this within the training scheme, a previous study [16] proposed a more generalized form of softmax that incorporates temperature T into the equation as in

$$q(z_i, T) = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \tag{1}$$

where z_i denotes the i-th logit in a neural network output layer and z_j includes the other remaining logits in the output. If T is larger than 1, the relative value between the probability of classes becomes smaller. Given the modified softmax function q_i, the distillation loss can therefore be formulated by measuring the difference between the teacher z_θ and student v_w output distibutions as in

$$L_D = \frac{1}{m}\sum_{i=1}^{m} KL\left(q\left(z_\theta\left(x^{(i)}\right), T\right), q\left(v_w\left(x^{(i)}\right), T\right)\right) \tag{2}$$

$$L = (\alpha - 1)L_D + \frac{\alpha}{m}\sum_{i=1}^{m}\sum_{k=1}^{C} y_c^{(i)} \log\left(q\left(v_{w,c}\left(x^{(i)}\right), 1\right)\right) \tag{3}$$

where the term added to the distillation loss L_D is the cross-entropy loss of the student, $y_c^{(i)}$ is the i-th label, and α is the portion that the cross-entropy contributes to the cost function. Given that the cross-entropy uses the original softmax, temperature T in this portion is always set to 1.

2.1. **Proposed teacher network.** This study implemented a similar HAR-architecture idea to that in [18] in constructing the teacher network. Specifically, the architecture of the teacher network z_θ proposed in this study combines the inception module with residual skip connection adapted from an image recognition model architecture proposed by [19]. In order to fit such an architecture to extract the features from 1D signal data, we replaced each kernel with a 1D convolution kernel and convolved it to each of the 1D accelerometer inputs. The proposed inception block consists of three main computational paths: convolutions with a kernel size of 1 followed by a kernel size of 3; convolutions with a kernel size of 1 followed by a kernel size of 5; a 1D average pooling operation followed by convolution with a kernel size of 1. To improve generalization to unseen examples, dropout [20] was implemented at each end of the path. The three computational paths were then concatenated by stacking each channel on top of another. At the end of the module, a residual connection was added to skip the computation performed by the inception module, as illustrated by the dashed line in Figure 1.

Given the inception-resnet block in Figure 1, the overall architecture was designed by stacking these modules four times and following them with a fully connected dense output layer of 11 outputs corresponding to the number of activities in the Skoda Dataset. The complete architecture of the network is depicted in Figure 2. With this architecture, the total number of learnable parameters contained in this network is 1,707,979.

2.2. **Student network.** The student network utilized in this study is a simple 1D-convolutional neural network that consists of two convolutional layers, each with a kernel size of 5, batch normalization, ReLU activation functions, and dropout. Each convolutional layer is then followed by average pooling with a kernel size and stride of 2. The last convolution layer is then fed into a dense output with 11 classes of human activities from the Skoda Dataset. Given the 69,579 learnable parameters contained in this model, the student is therefore 24.5 times smaller than the teacher.

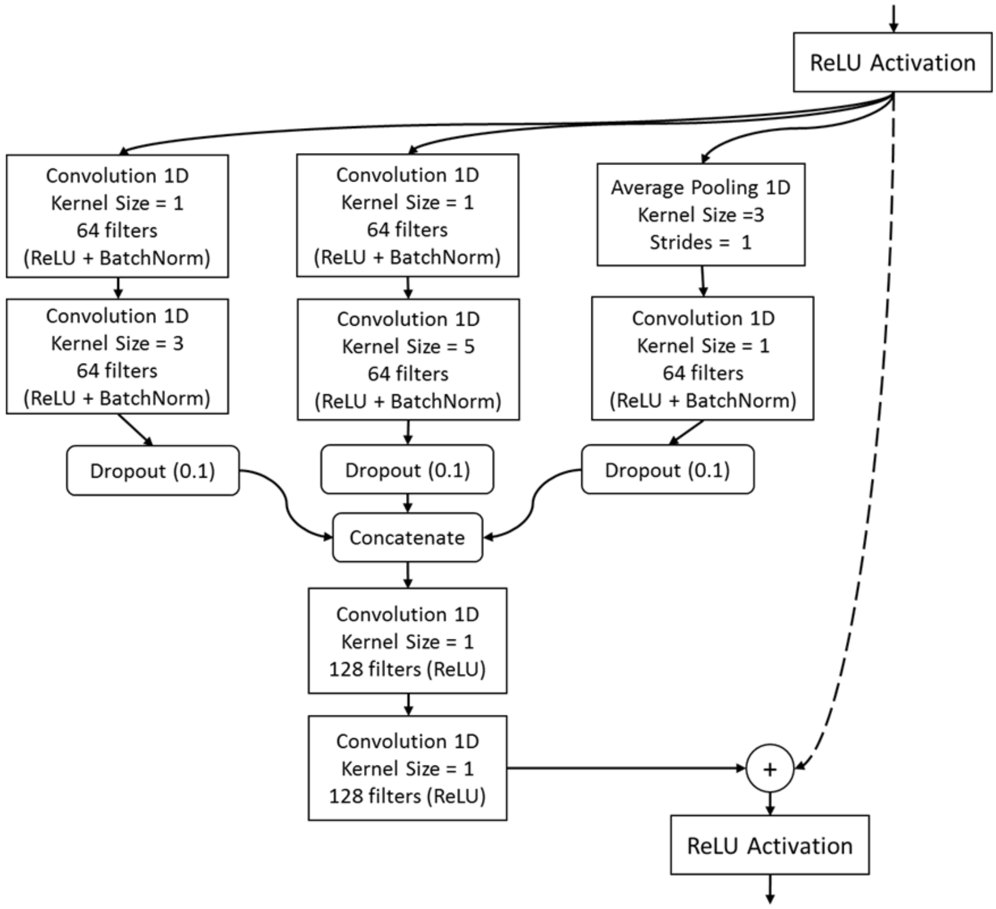

FIGURE 1. Proposed inception-resnet module with 1D convolution

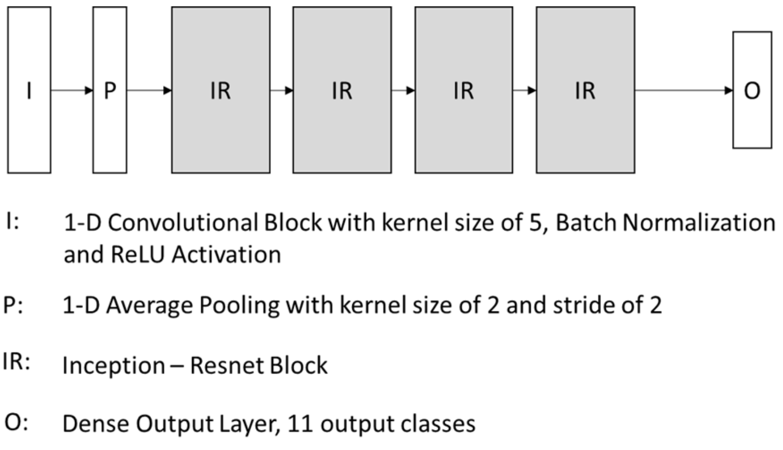
I
P
IR
IR
IR
IR
O
I: 1-D Convolutional Block with kernel size of 5, Batch Normalization and ReLU Activation
P: 1-D Average Pooling with kernel size of 2 and stride of 2
IR: Inception – Resnet Block
O: Dense Output Layer, 11 output classes

FIGURE 2. Proposed inception-resnet architecture as teacher network

3. **Experimental Setup and Results.** To train and evaluate the HAR model and perform knowledge distillation on the Skoda Dataset, a series of experiments were conducted in sequential steps: data preprocessing, training and evaluation of the teacher model, and distillation and evaluation of the student model. We performed these experiments on a computer with an Intel(R) Core(TM) i7-4790K CPU of 4.0 GHz and 30 GB RAM along with an NVIDIA GTX 1080 Ti Graphic card for the tensor operations.

3.1. **Data preprocessing.** The Skoda Dataset used in this study was obtained by placing accelerometers on 10 different positions on each subject's arms. To accommodate supervised classification task, 10 manipulative gestures activities and a null activity were labeled in each timestep of the sampled sequence [17]. Given the raw time-series data, this study implemented preprocessing steps similar to those in [8] by representing each example $x^{(i)}$ in the training set X as a window sliced across the signal, as illustrated in Figure 3. Therefore, the size of each example was $x^{(i)} \in \mathbb{R}^{w \times n}$, and the dataset size was $X \in \mathbb{R}^{m \times w \times n}$, where m is the number of examples on the dataset, w is the size of the window, and n is the number of features, which represents 60 sensor readings for each of the accelerometers in the x, y, and z directions. In order to compare the performance of the present model with that in [8], we downsampled the signals to approximately 30 Hz and sliced the window according to a 50% overlap. With the same random seed as in [8], the dataset was then divided, using 80% of the examples as the training set and 10% each for the validation and test sets.

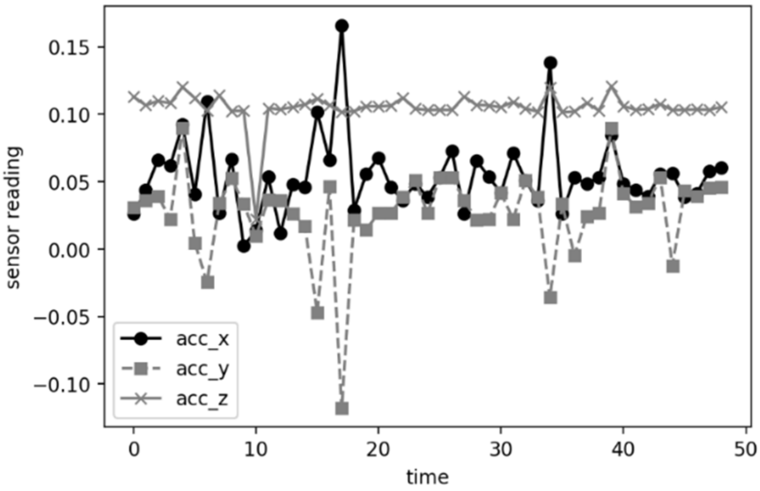

Figure 3. Window sliced over acceleremetor sensor reading

3.2. **Training and evaluating of teacher model.** In order to update the weights at each gradient descent step, the Nadam optimizer [21] with a learning rate of 0.001 was utilized. We trained the teacher model with a batch size of 64 over 400 epochs as shown in Figure 3. The minimization of the cross-entropy loss resulted in what is plotted in Figure 4, which indicates a convergence at the end of the training with the accuracy similarly converged around the same epochs. The best validation accuracy during training, 99.2%, was obtained at epoch 313. The chosen teacher model in this study, therefore, used the learning parameters obtained during this epoch.

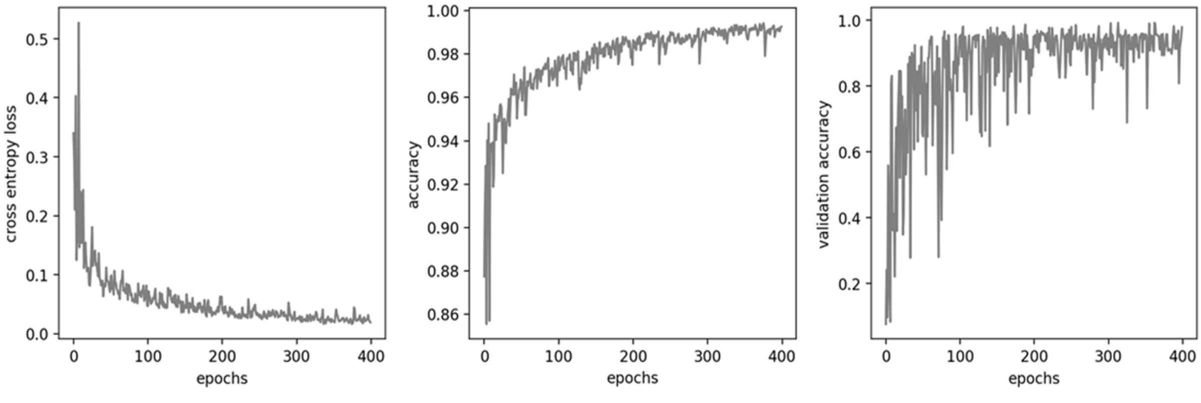

Figure 4. Convergence of cross-entropy loss during training

The teacher model was further evaluated with the test set by using the macro-averaged F1-score metrics. We then obtained the score of 97.7%, which is a very comparable performance to that achieved by [8], which yielded a 97% score on the same test set.

3.3. **Performing distillation training and evaluating student model.** The distillation training was conducted by back-propagating the gradients of the cost function (3) with respect to the student model weight parameters w. We utilized a half-portion of the cross-entropy loss by setting α to 0.5 and temperature T to 3. The model was trained by using the Nadam optimizer with a learning rate of 0.0005 for 20 epochs with a batch size of 64. The results of the training are plotted in Figure 4. The student loss plotted in Figure 5 is the cross-entropy loss obtained by comparing the softmax predictions with the labels, and the distillation loss is the KL-divergence between the teacher and student models.

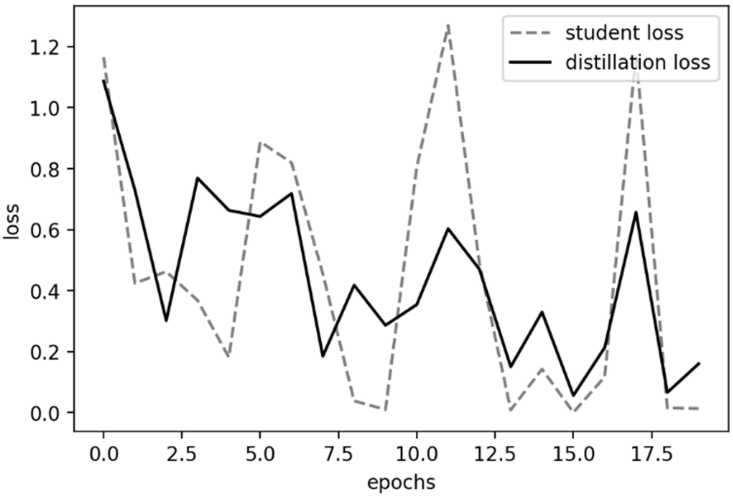

Figure 5. Knowledge distillation training

To further compare the performance of the distillation training to the regular supervised training, we trained a baseline CNN model with the same architecture and hyperparameter settings as the student model. The CNN model obtained through the distillation training achieved a macro-averaged F1-score accuracy of 96.4%, while the baseline CNN trained from scratch achieved an accuracy of only 86.4%. To clearly show the comparison results, the F1-scores of the models are further summarized in Table 1.

Table 1. Test F1-score accuracy on the Skoda Dataset

Model	Distillation scheme	F1-score
Self-attention model [8]	N/A	97%
Baseline CNN	N/A	86.4%
HAR inception-resnet	Teacher network	97.7%
Distilled CNN	Student network	96.4%

4. **Conclusion.** In this paper, we proposed a lightweight HAR model that uses the Skoda Dataset baseline to distill the knowledge of a deeper teacher model into a simple CNN. The teacher model architecture designed in this study is a deep neural network consisting of inception-residual modules with a total number of 1,707,979 learnable parameters. It achieved a macro-averaged F1-score of 97.7%, which is very comparable to the reported F1-score of the state-of-the-art self-attention model in [8], with its score of 97%. The distillation of the teacher model into a CNN training model resulted in a score of 96.4%.

Comparing this performance to the baseline CNN that was trained from scratch, the distilled-CNN yielded a 10% increase of macro-averaged F1-score. By using knowledge distillation training, this study was therefore able to obtain a simpler, lightweight model offering HAR performance close to that of the state-of-the-art [8] and teacher models. As the results of this paper indicate the effectiveness of knowledge distillation application to HAR, future potential extensions of using improved distillation methods such as teacher assistant or noisy student distillations [22,23] can be further investigated to close the performance gap between the teacher and the student models.

Acknowledgement. This work was supported by a National Research Foundation of Korea (NRF) grant (no. 2020R1A2C110229411) funded by the Korean government (MSIT).

REFERENCES

[1] A. Ferrari, D. Micucci, M. Mobilio and P. Napoletano, Hand-crafted features vs residual networks for human activities recognition using accelerometer, *2019 IEEE 23rd Int. Symp. Consum. Technol. (ISCT2019)*, pp.153-156, doi: 10.1109/ISCE.2019.8901021, 2019.

[2] F. J. Ordóñez and D. Roggen, Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition, *Sensors (Switzerland)*, vol.16, no.1, doi: 10.3390/s16010115, 2016.

[3] G. Ogbuabor and R. La, Human activity recognition for healthcare using smartphones, *ACM Int. Conf. Proceeding Ser.*, pp.41-46, doi: 10.1145/3195106.3195157, 2018.

[4] B. Fu, F. Kirchbuchner and A. Kuijper, Performing realistic workout activity recognition on consumer smartphones, *Technologies*, vol.8, no.4, p.65, doi: 10.3390/technologies8040065, 2020.

[5] L. C. Günther, S. Kärcher and T. Bauernhansl, Activity recognition in manual manufacturing: Detecting screwing processes from sensor data, *Procedia CIRP*, vol.81, pp.1177-1182, doi: 10.1016/j.procir.2019.03.288, 2019.

[6] V. S. Murahari and T. Plotz, On attention models for human activity recognition, *Proc. – Int. Symp. Wearable Comput. (ISWC)*, pp.100-103, doi: 10.1145/3267242.3267287, 2018.

[7] C. Xu, D. Chai, J. He, X. Zhang and S. Duan, InnoHAR: A deep neural network for complex human activity recognition, *IEEE Access*, vol.7, pp.9893-9902, doi: 10.1109/ACCESS.2018.2890675, 2019.

[8] S. Mahmud et al., Human activity recognition from wearable sensor data using self-attention, *Front. Artif. Intell. Appl.*, vol.325, pp.1332-1339, doi: 10.3233/FAIA200236, 2020.

[9] A. Bayat, M. Pomplun and D. A. Tran, A study on human activity recognition using accelerometer data from smartphones, *Procedia Comput. Sci.*, vol.34, pp.450-457, doi: 10.1016/j.procs.2014.07.009, 2014.

[10] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou and Y. Amirat, Physical human activity recognition using wearable sensors, *Sensors (Switzerland)*, vol.15, no.12, pp.31314-31338, doi: 10.3390/s151229858, 2015.

[11] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu and Y. Liu, Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities, *arXiv.org*, arXiv: 2001.07416, 2020.

[12] J.-Y. Zhao, J. Gong, S.-T. Ma and Z.-M. Lu, Curvature gray feature decomposition based finger vein recognition with an improved convolutional neural network, *International Journal of Innovative Computing, Information and Control*, vol.16, no.1, pp.77-90, doi: 10.24507/ijicic.16.01.77, 2020.

[13] H. Haresamudram, D. V. Anderson and T. Plötz, On the role of features in human activity recognition, *Proc. – Int. Symp. Wearable Comput. (ISWC)*, pp.78-88, doi: 10.1145/3341163.3347727, 2019.

[14] M. Tan and Q. V. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, *The 36th Int. Conf. Mach. Learn. (ICML2019)*, pp.10691-10700, 2019.

[15] T. B. Brown et al., Language models are few-shot learners, *arXiv.org*, arXiv: 2005.14165, 2020.

[16] G. Hinton, O. Vinyals and J. Dean, Distilling the knowledge in a neural network, *arXiv.org*, arXiv: 1503.02531, 2015.

[17] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz and G. Tröster, Wearable activity tracking in car manufacturing, *IEEE Pervasive Comput.*, vol.7, no.2, pp.42-50, doi: 10.1109/MPRV.2008.40, 2008.

[18] M. Ronald, A. Poulose and D. S. Han, ISPLInception: An Inception-ResNet deep learning architecture for human activity recognition, *IEEE Access*, vol.9, pp.68985-69001, doi: 10.1109/ACCESS.2021.3078184, 2021.

[19] C. Szegedy, S. Ioffe, V. Vanhoucke and A. A. Alemi, Inception-v4, Inception-ResNet and the impact of residual connections on learning, *The 31st AAAI Conf. Artif. Intell.*, pp.4278-4284, 2017.

[20] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.*, vol.15, pp.1929-1958, doi: 10.1109/CVPR.2018.00797, 2014.

[21] T. Dozat, Incorporating Nesterov momentum into Adam, *ICLR Work.*, no.1, pp.2013-2016, 2016.

[22] S.-I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa and H. Ghasemzadeh, Improved knowledge distillation via teacher assistant, *arXiv.org*, arXiv: 1902.03393, 2019.

[23] Q. Xie, M.-T. Luong, E. Hovy and Q. V. Le, Self-training with noisy student improves ImageNet classification, *arXiv.org*, arXiv: 1911.04252, 2019.